CS 8803-MDS
Human-in-the-loop Data Analytics

Lecture 11

09/28/22

Today’s class

Benchmarking Spreadsheet Systems

Author: Abhinav, Harshal

Reviewer: Qiandong

Archaeologist: Cuong

Practioner: Ting

PORTLAND

ACM SIGMOD/PODS International Conference on Management of Data
June 14-19th, 2020

ACM SIGMOD +PODS 2020

Portland, Oregon USA

SIGMOD PODS

ACM SIGMOD/PODS International Conference on Management of Data June 14 - June 19, 2020 Portland, OR, USA

Welcome

- Homepage
- News

Organization

- Experience Report
- Organization
- SIGMOD PC
- PODS PC

2020 ACM SIGMOD/PODS @ Portland, OR, USA

SIGMOD/PODS 2020 Experience Report now available

Coronavirus Updates

Updates on SIGMOD/PODS 2020 Registration (12 May 2020)

Calls For Submissions

- Important Dates
- Calls for Submissions

PODS Program

- Program Overview
- Detailed Program
- Research Papers
- Keynote Talk

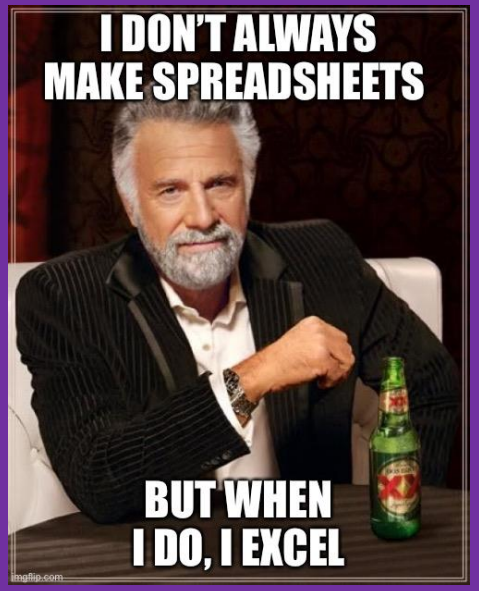

CAUSE OF DEATH
MICROSOFT EXCEL
VIA 9GAG.COM

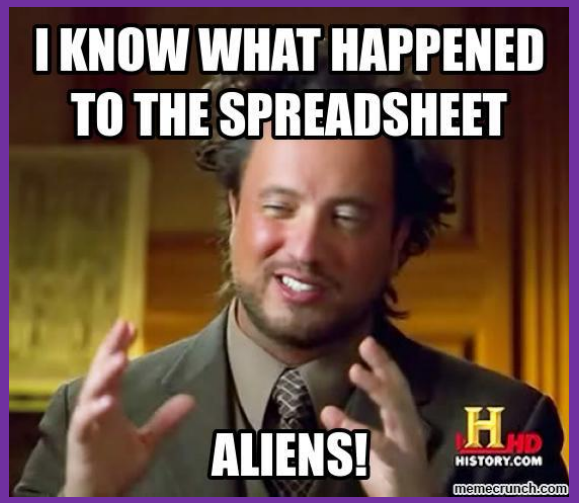

I LOVE MICROSOFT EXCEL
BECAUSE IT MAKES PEOPLE CRY!


SORRY I BOTHERED YOU
WITH MY AMAZING SPREADSHEET

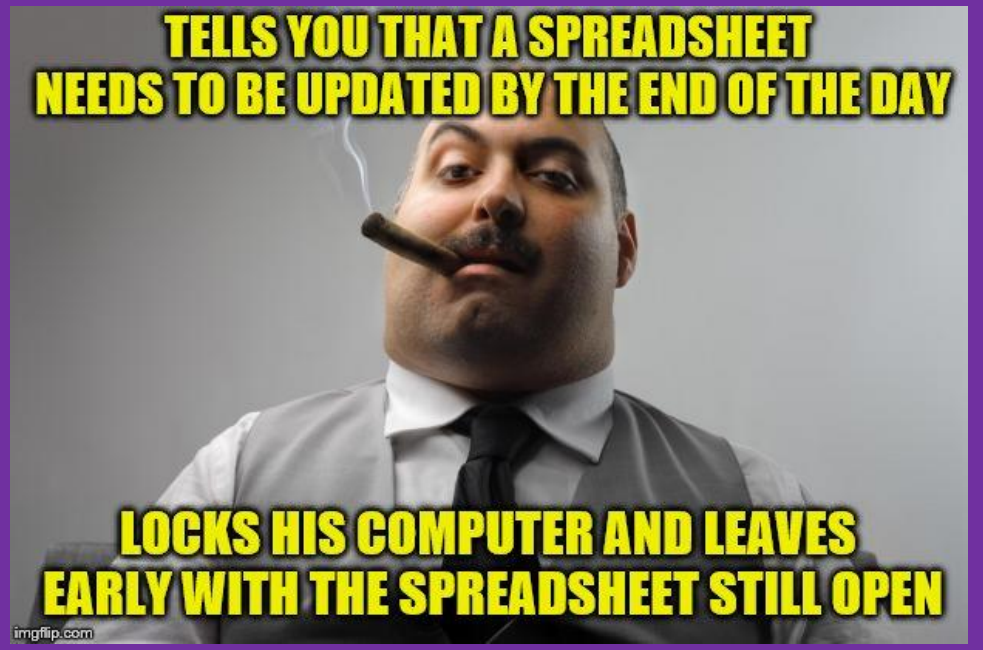

WE'RE GONNA NEED A BIGGER
SPREADSHEET
memegenerator.net

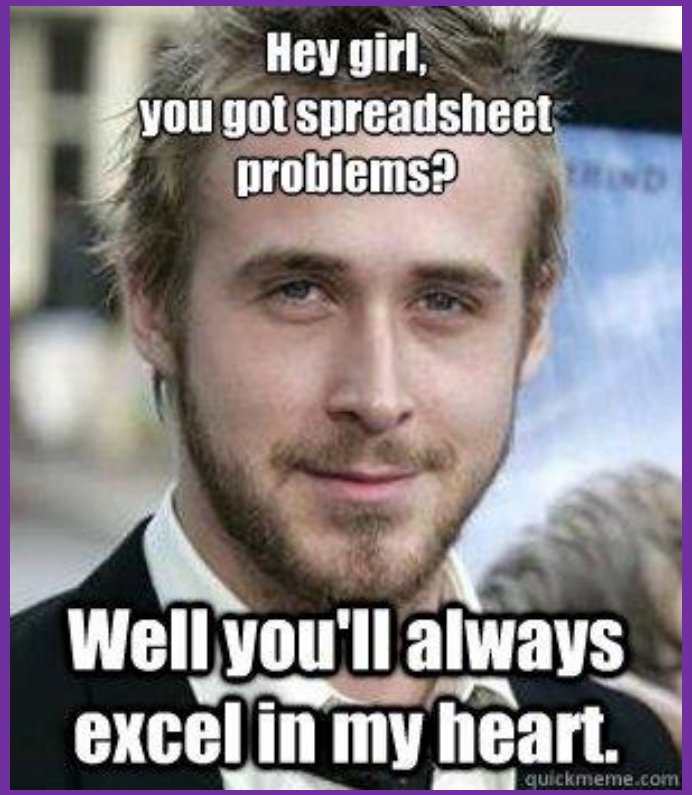

I STARTED THE DAY WITH LOTS OF PROBLEMS.
BUT NOW, AFTER HOURS AND HOURS OF WORK,
I HAVE LOTS OF PROBLEMS IN A SPREADSHEET.


I DON'T ALWAYS MAKE SPREADSHEETS
BUT WHEN I DO, I EXCEL
imgflip.com


I KNOW WHAT HAPPENED TO THE SPREADSHEET
ALIENS!
HISTORY.COM
memecrunch.com


LOOK AT ALL THE COLORS
ON MY SPREADSHEET!
imgflip.com


TELLS YOU THAT A SPREADSHEET NEEDS TO BE UPDATED BY THE END OF THE DAY
LOCKS HIS COMPUTER AND LEAVES EARLY WITH THE SPREADSHEET STILL OPEN
imgflip.com

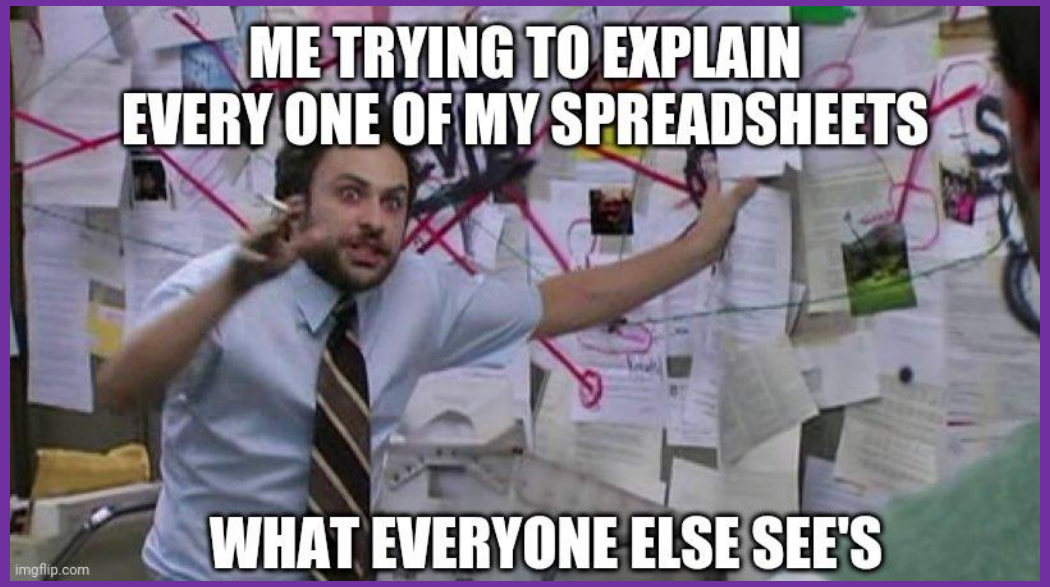

Hey girl,
you got spreadsheet problems?
Well you'll always excel in my heart.
quickmeme.com

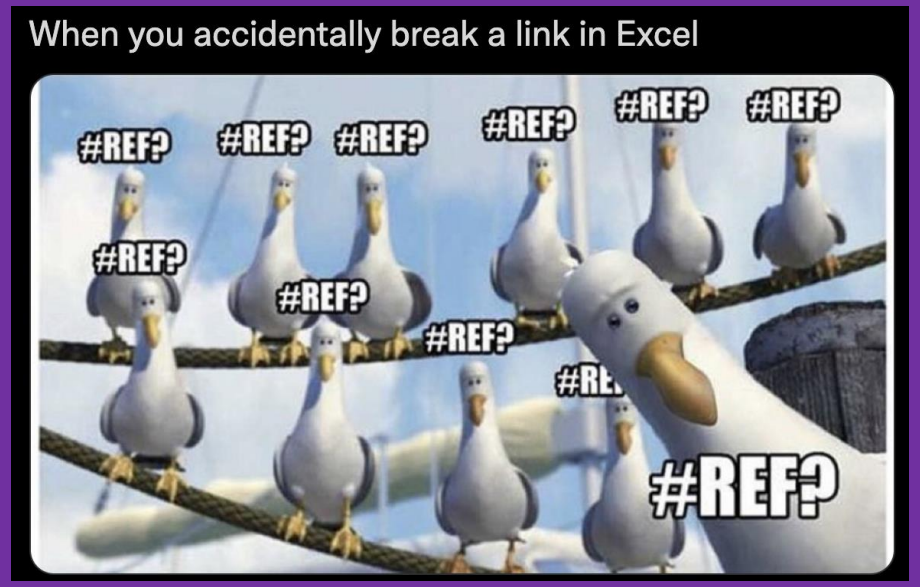

NO MORE
SPREADSHEET
makeameme.org

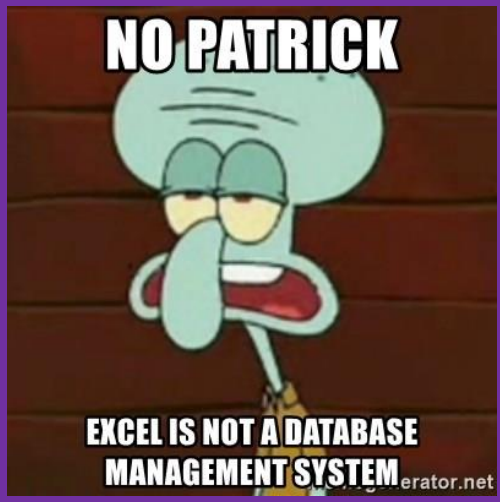

NO MORE SPREADSHEETS...
PLEASE
makeameme.org


BEHOLD
EXCEL SPREADSHEET EVERYWHERE
makeameme.org

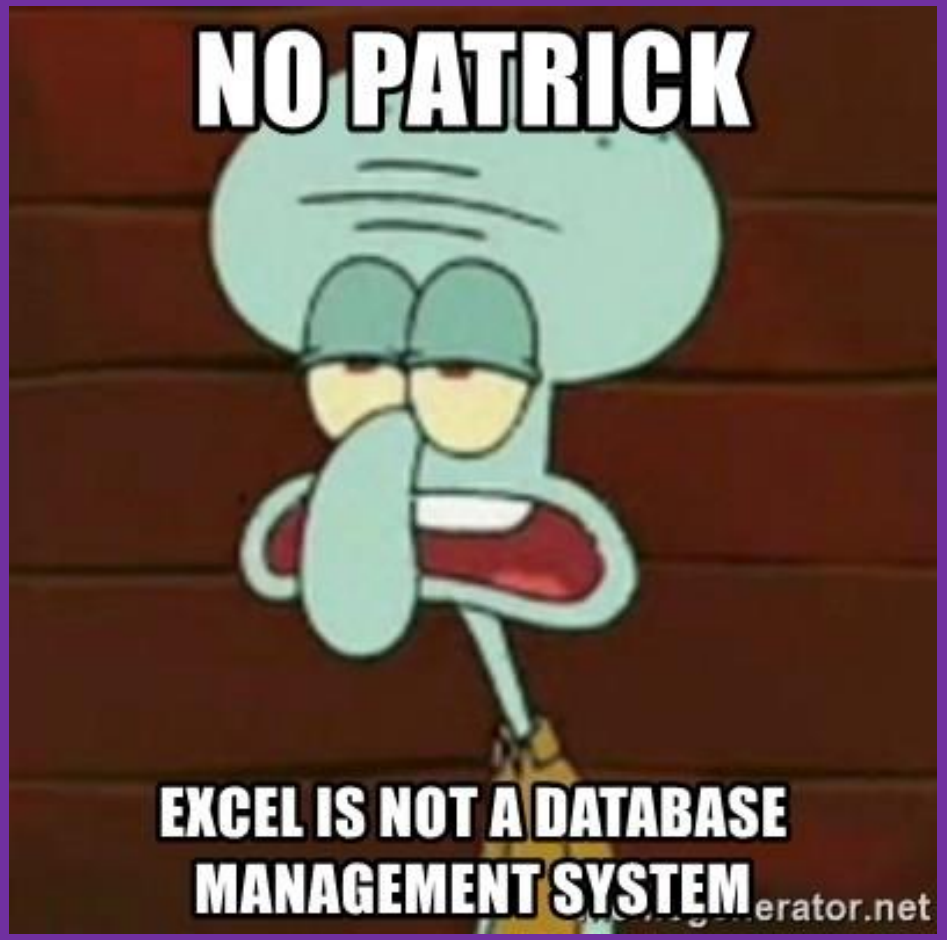

Me:
12
5
12.5
Excel:
12
5
1/12/1900
12:00:00 PM


ME TRYING TO EXPLAIN
EVERY ONE OF MY SPREADSHEETS
WHAT EVERYONE ELSE SEE'S
imgflip.com


When you accidentally break a link in Excel
#REF?
#REF?
#REF?
#REF?
#REF?
#REF?
#REF?
#REF?
#REF?
#REF?


NO PATRICK
EXCEL IS NOT A DATABASE MANAGEMENT SYSTEM
erator.net


BEHOLD
EXCEL SPREADSHEET
EVERYWHERE
makeameme.org


NO PATRICK
EXCEL IS NOT A DATABASE
MANAGEMENT SYSTEM

Serious* Investigation Needed

*very very

SUSPECTS

- Unfortunately, originality of colors doesn’t help

TEST-1

- Basic Complexity Testing (BCT)
  - Test basic operations
    - Opening
    - Structuring
    - Editing
    - Analyzing data
  - Goal is to understand the impact of
    - Type of operation
    - Size of data
  - Measure when the systems become dangerous for interactive workload

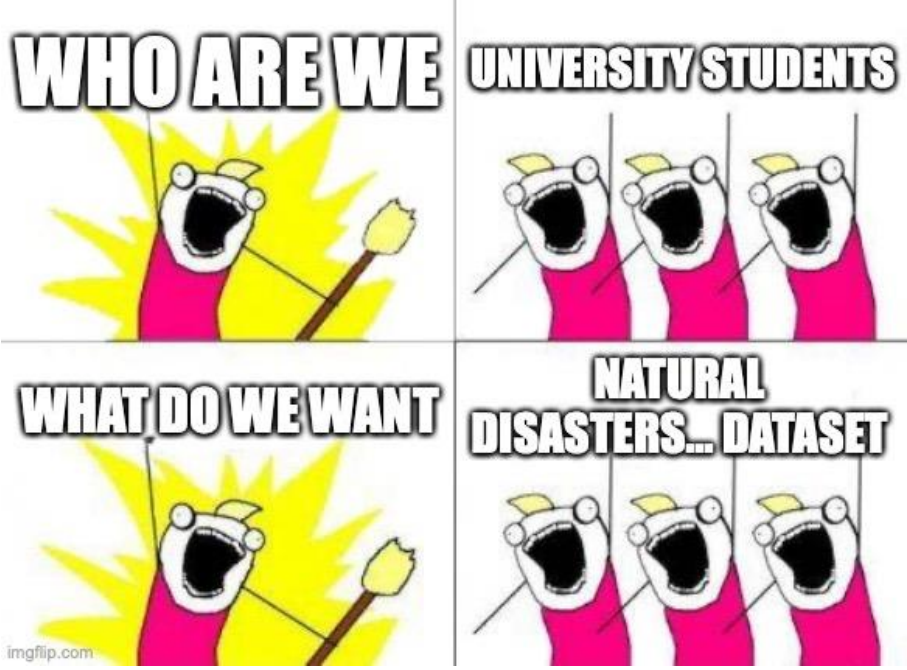
TEST - 2

- Optimization Opportunities Testing (OOT)
  - Learn about whether spreadsheet developers use latest academic research
    - Create indexes
    - Incremental updates
    - Workload aware data layout
    - Sharing of computation
  - Goal is to identify new opportunities for improving the design to support computation on large datasets and ofc save lives

TESTS SPECIFIC DETAILS

- Addressing Interaction effects
  - Problem: **Any change on the spreadsheet will lead to additional formulae re-computation**
  - Solution: We operate on real-world datasets containing both formulae and raw data, as well as datasets with raw data only.
- Addressing Human errors
  - Problem: **Human error and maintain repeatability**
  - Solution: VBA (VB for Apps) for Excel, Calc Basic for Calc, GAS (Google Apps Script)
- Coverage
  - BCT: Classified operations into 7 classes based on complexity & type of input
  - OOT: We relied on our creativity to find settings where ∃ DB-like optimizations

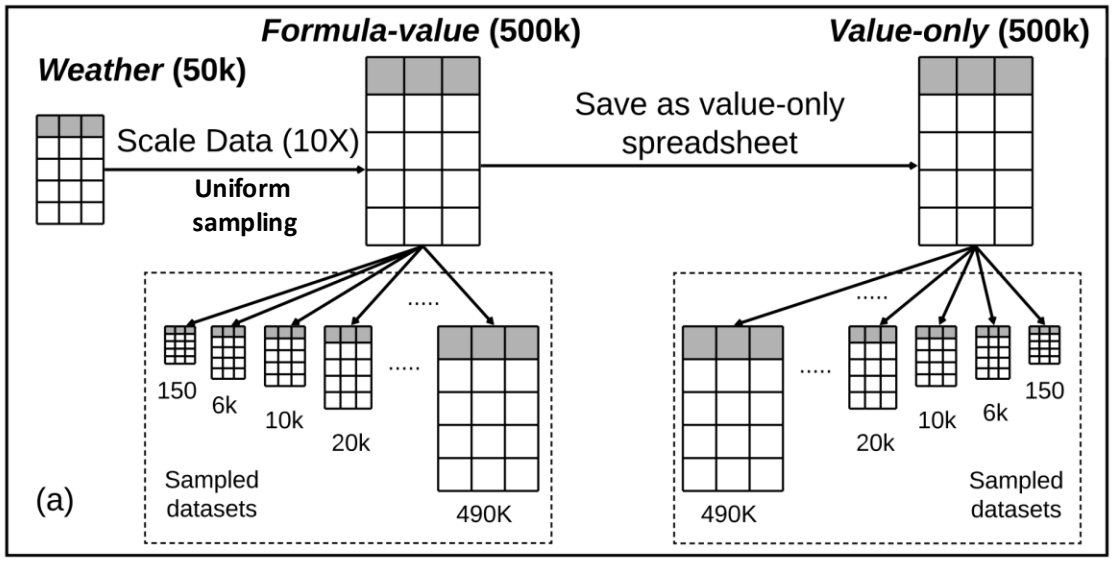
DATASET



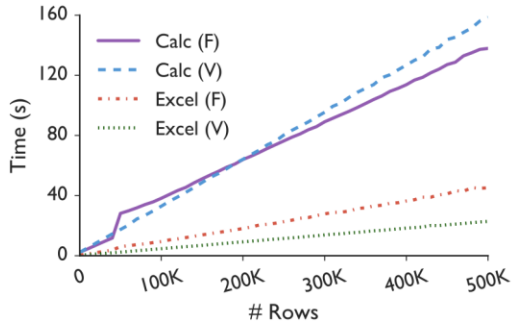
1. A spreadsheet on weather data across the states in US, containing 50000 rows and 17 columns
2. Cells within seven of those columns contained COUNTIF formulae which
   - will output 0 or 1 depending on the previous column and same row cell

Type	isStorm?	isEarthquake?
Storm	1	0
Earthquake	0	1

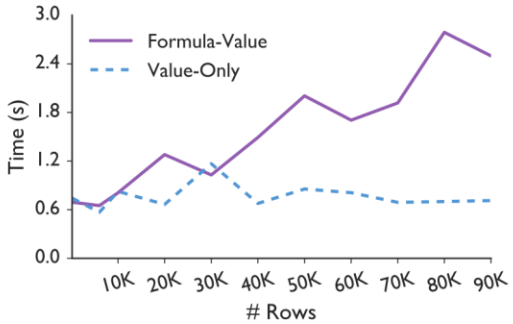
3. Using this dataset as the starting point, we create:



TEST 1 – BCT (Basic Complexity Testing) (1/4)

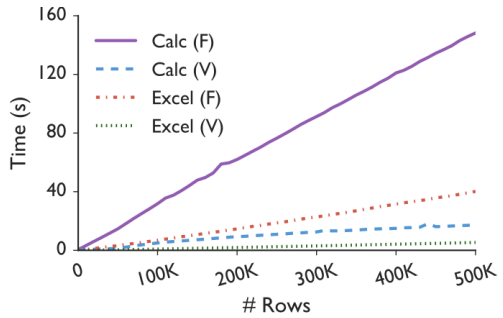


(a) Excel and Calc

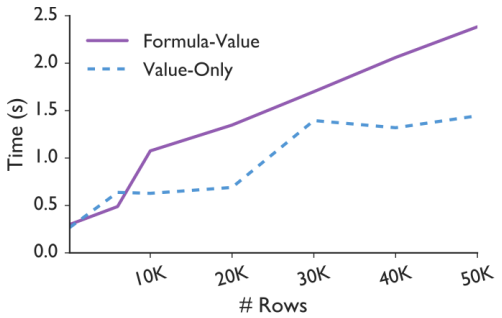


(b) Google Sheets

Figure 2: *Open* in Excel, Calc is slow; it is faster on Google Sheets due to lazy loading of data not in the user window.



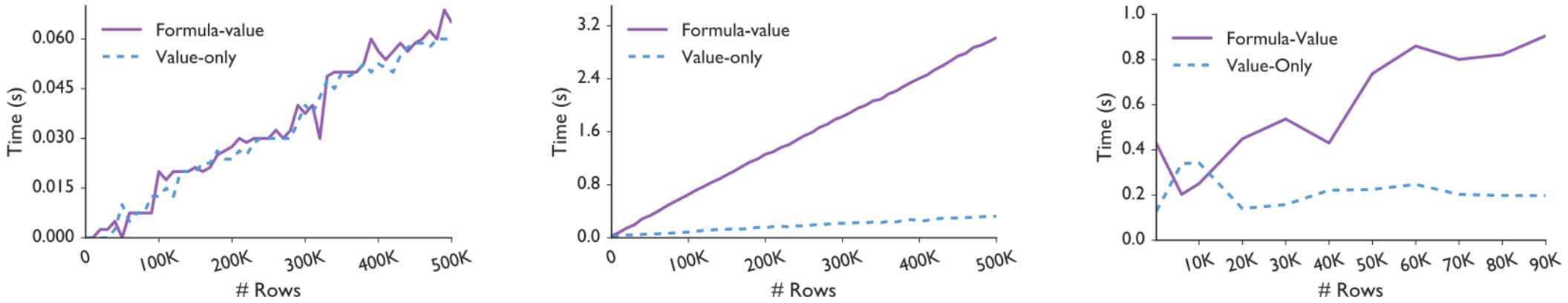
(a) Excel and Calc



(b) Google Sheets

Figure 3: Sort on Formula-value is substantially worse than Value-only, thanks to formula recomputation on sort.

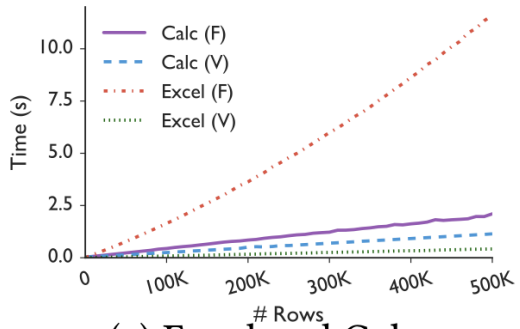
TEST 1 – BCT (Basic Complexity Testing) (2/4)



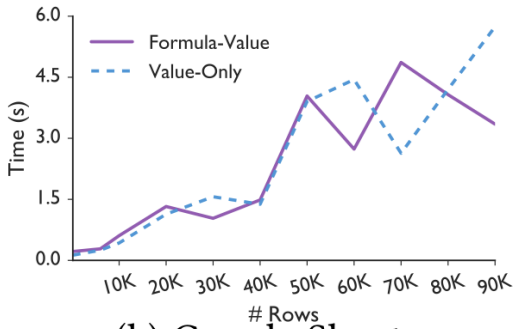
(a) Excel (b) Calc (c) Google Sheets

Figure 4: While *conditional formatting* on Formula-value is slow for Calc and Google Sheets due to formula recomputation, no such recomputation is triggered in Excel. Google Sheets is faster for Value-only due to formatting cells in a lazy fashion.

TEST 1 – BCT (Basic Complexity Testing) (3/4)

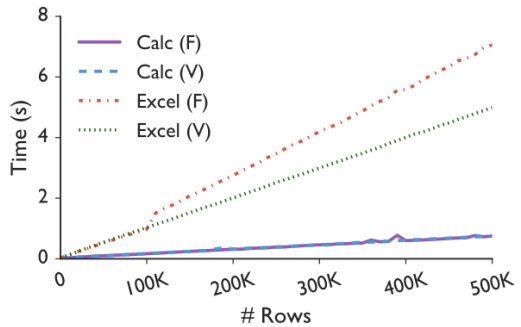


(a) Excel and Calc

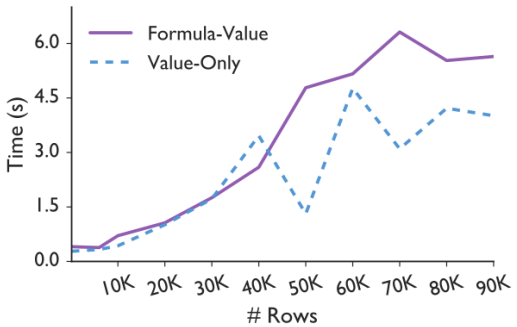


(b) Google Sheets

Figure 5: *Filter* on Formula-value in Excel does unnecessary recomputation. Google Sheets is slower than the other two.



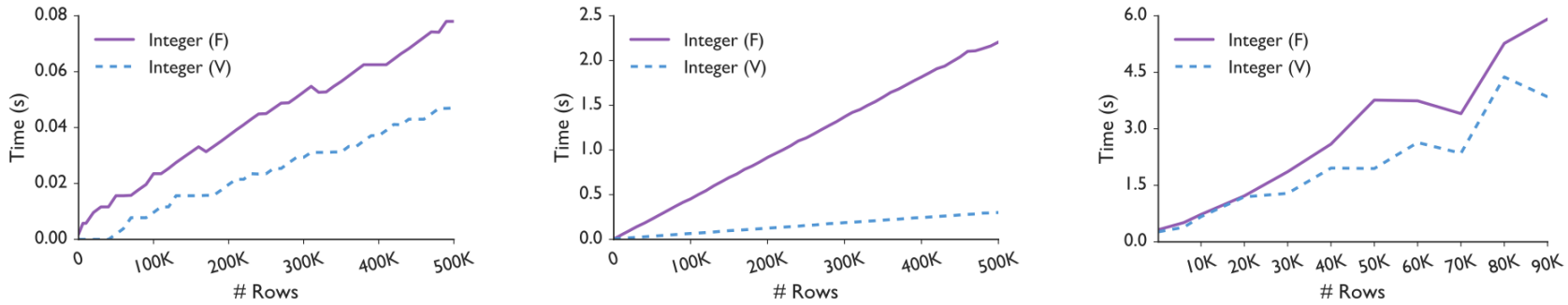
(a) Excel and Calc



(b) Google Sheets

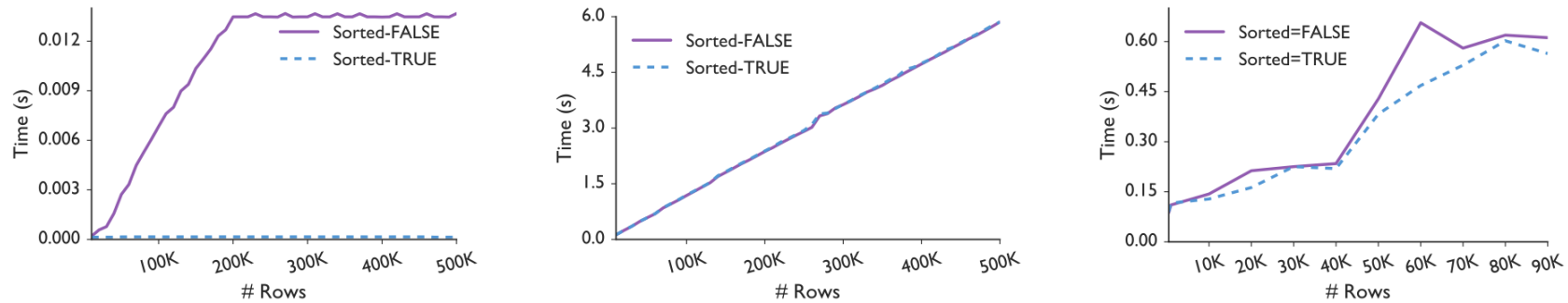
Figure 6: Calc is faster than the other two for Pivot Tables

TEST 1 – BCT (Basic Complexity Testing) (4/4)



(a) Excel (b) Calc (c) Google Sheets

Figure 7: COUNTIF **is extremely fast in Excel compared to Calc and Google Sheets. However, for both Excel and Calc, latency is higher in Formula-value due to formula recomputation.**



(a) Excel (b) Calc (c) Google Sheets

Figure 8: For VLOOKUP**, while Excel terminates after finding a matching value, Calc and Google Sheets continue to scan the entire data. Excel optimizes approximate search (Sorted=True) via an efficient searching algorithm,** *e.g.*, **binary search.**

TEST 1 – BCT Summary

	Formula-value			Value-only		
	E (%)	**C** (%)	**G** (%)	**E** (%)	**C** (%)	**G** (%)
Open	**0.6**	0.015	0.05	**0.6**	0.015	0.05
Sort	1	0.6	**3.4**	**7**	1	2.04
Conditional Formatting	**100**	8	17	**100**	**100**	**100**
Filter	4	**12**	3.4	**100**	20	6.8
Pivot Table	5	**34**	3.4	5	**33**	6.8
COUNTIF	**100**	11	3.4	**100**	**100**	3.4
VLOOKUP	×	×	×	**100**	5	23.8

Test 2: Optimisation Opportunities Testing

- Do current spreadsheet use standard database optimizations?
  - Indexing?
  - Columnar data layout?
  - Shared computation?
  - Eliminating redundant computation?
  - Incremental updates?

Indexing?

- COUNTIF(Where do you want to look?, What do you want to look for?)
  - How many cells in a column contain 1?
  - If there was an index on this column, this would have been near constant time.
  - But all three spreadsheets took linear time!
- VLOOKUP(what to look for?, where?, which column to output, approx?)
  - Query such that a linear scan would go through 200k rows
  - If there was an index, O(logn)
  - But all three spreadsheets took linear time*

Indexing?

- **Find and replace**
  - Inverted index?
  - A mapping between content -> location

Carolina	K1	K2		
South	K1		K3	
North		K2		K4
Dakota			K3	K4

Indexing

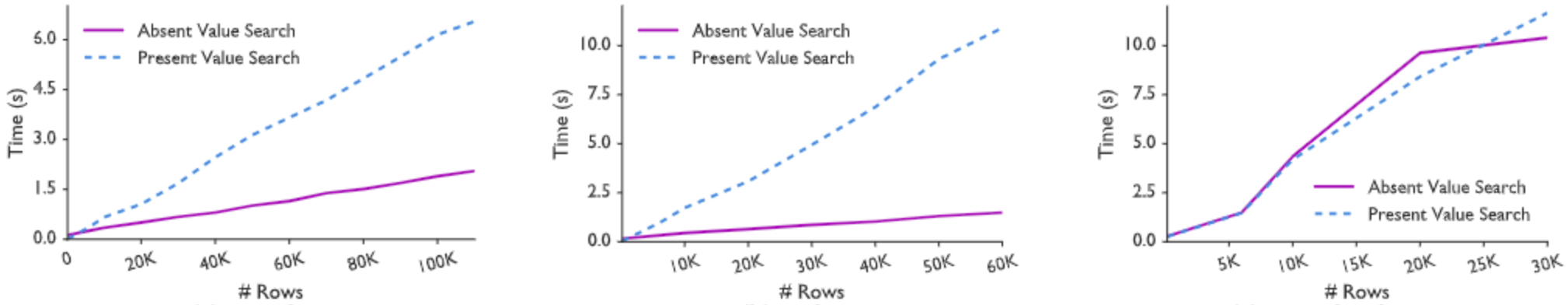
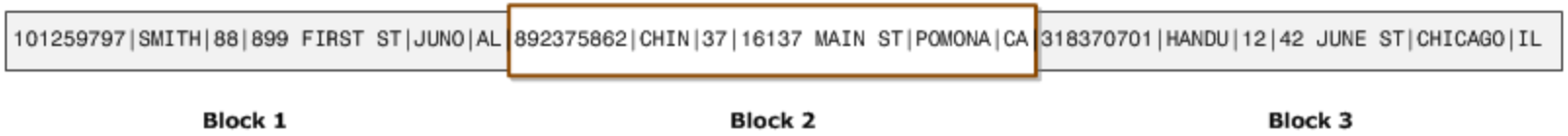


Figure 9: A linear trend for *Find and Replace* indicates the absence of an index.

In memory data layout?

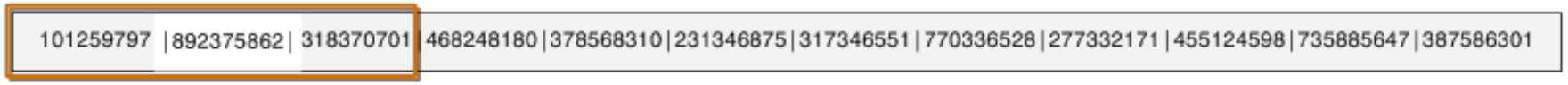
SSN	Name	Age	Addr	City	St
101259797	SMITH	88	899 FIRST ST	JUNO	AL
892375862	CHIN	37	16137 MAIN ST	POMONA	CA
318370701	HANDU	12	42 JUNE ST	CHICAGO	IL



Source: https://docs.aws.amazon.com/redshift/latest/dg/c_columnar_storage_disk_mem_mgmnt.html

In memory data layout

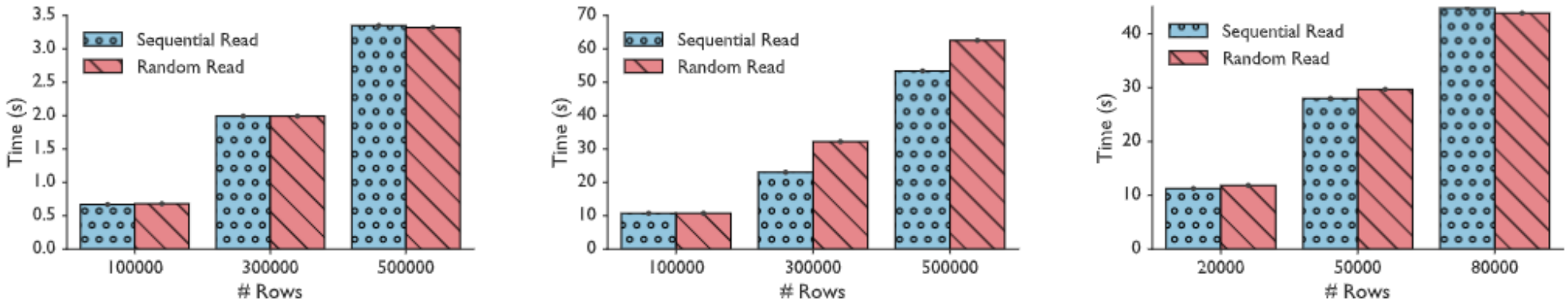
SSN	Name	Age	Addr	City	St
101259797	SMITH	88	899 FIRST ST	JUNO	AL
892375862	CHIN	37	16137 MAIN ST	POMONA	CA
318370701	HANDU	12	42 JUNE ST	CHICAGO	IL



Block 1

Source: https://docs.aws.amazon.com/redshift/latest/dg/c_columnar_storage_disk_mem_mgmnt.html

In memory data layout



(a) Excel (b) Calc (c) Google Sheets

Figure 10: For all three systems, sequential and random access of a column takes roughly the same time indicating the absence of a columnar data layout.

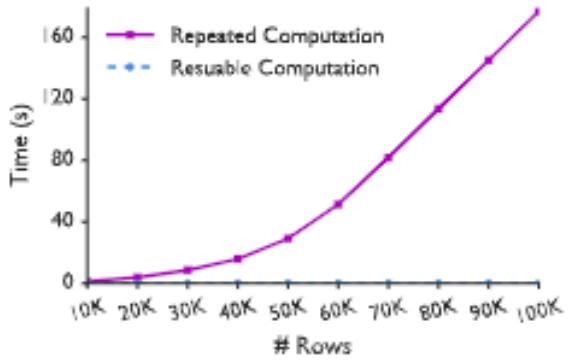
Spreadsheet systems do not employ a columnar data layout to improve computational (e.g., aggregation) performance!

Shared Computation?

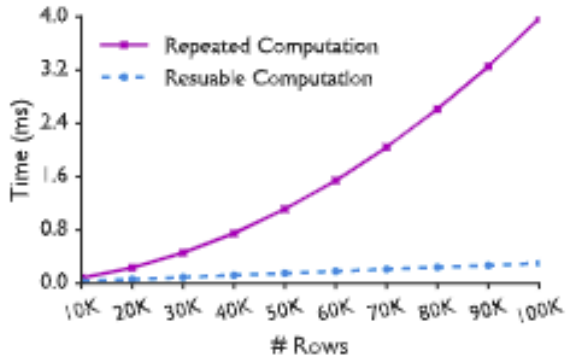
A	B	C
1	=SUM(A1:A1)	=A1
2	=SUM(A1:A2)	=A2+C1
3	=SUM(A1:A3)	=A3+C2
4	=SUM(A1:A4)	=A4+C3
• • •	• • •	• • •
n	=SUM(A1:An)	=An+C(n-1)

(a) Sample Data

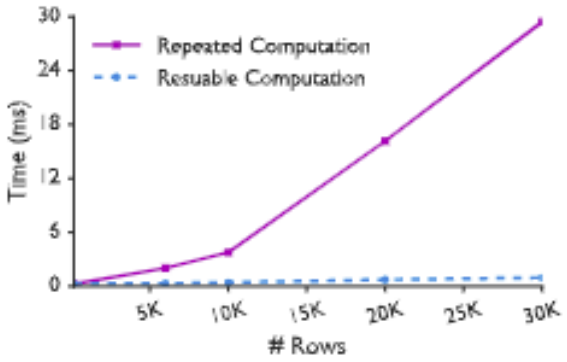
No Shared Computation.



(b) Excel



(c) Libre



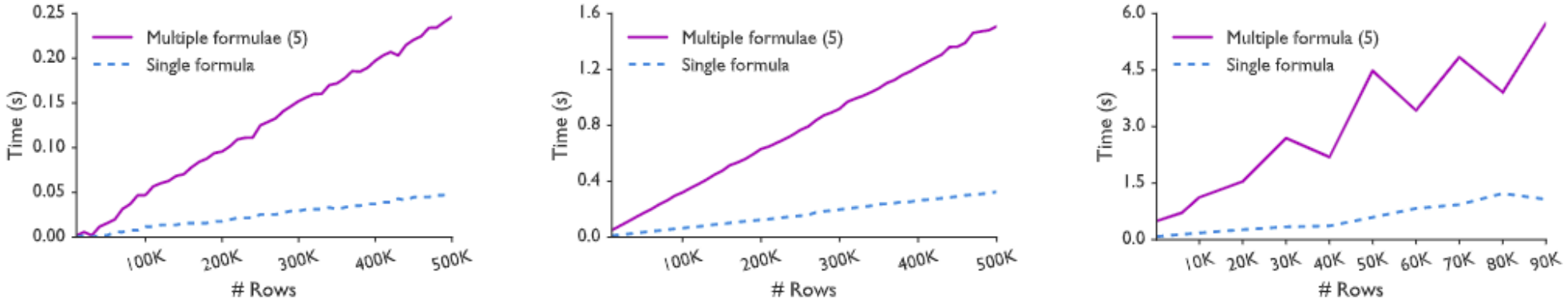
(d) GS

The systems were not able to detect sharing opportunities and use them to reduce computation!

Eliminating redundant computation?

- What if we made it easier to share results?
- Inserted five instances of the EXACT same formula
- This took 5x longer than when we inserted a single formula.

Redundant computation



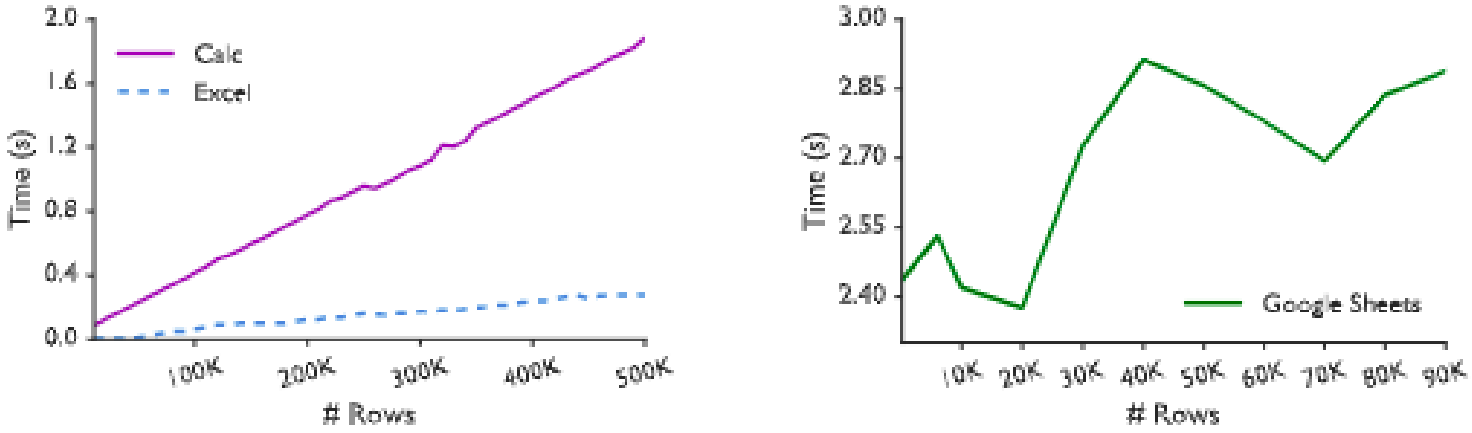
(a) Excel (b) Calc (c) Google Sheets

Figure 12: All three systems redundantly compute duplicate instances of a COUNTIF formula instead of reusing the previously computed result, causing the execution time to increase linearly with the number of duplicates.

Incremental updates

- COUNTIF(J1:Jm, 1)
- COUNTIF(J0:Jm, 1)
- Should be done in constant time if we support incremental updates.

But all of them recompute from scratch

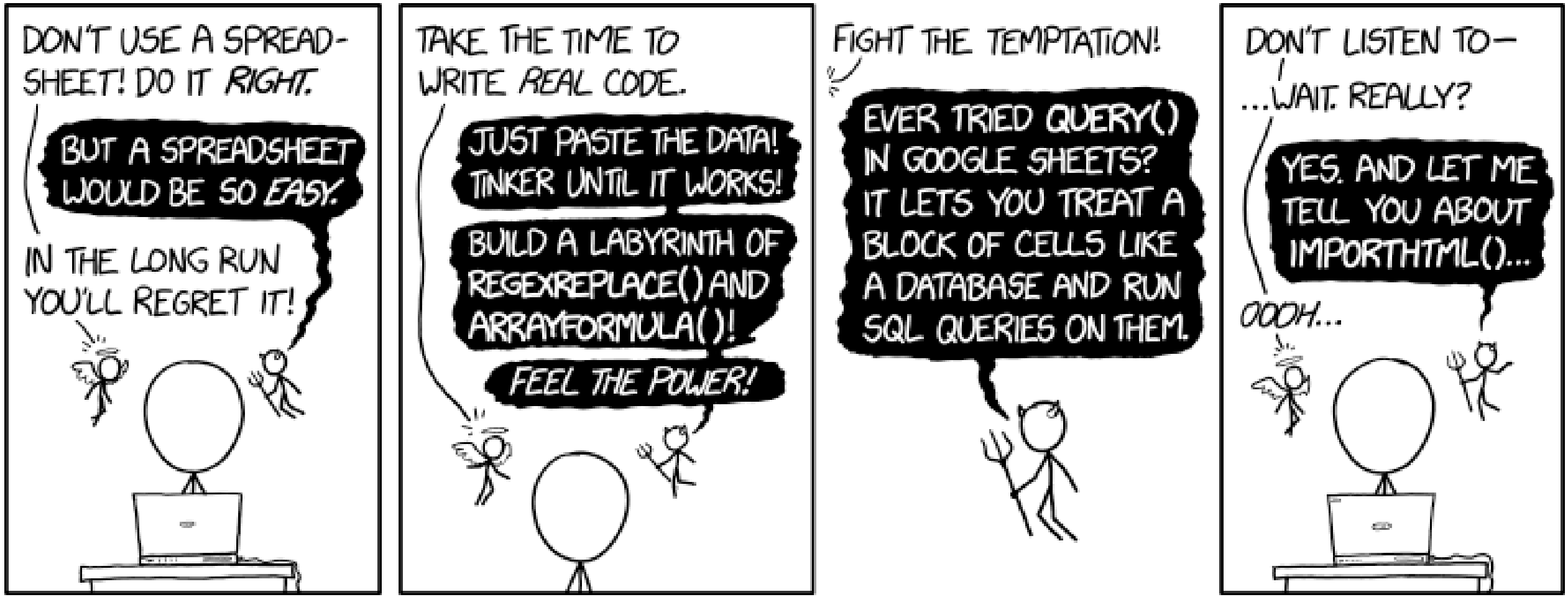


(a) Excel and Calc (b) Google Sheets

Figure 13: All three systems recompute the results of a COUNTIF formula from scratch after a single cell update.

In conclusion

- Indexing and data layout
- Shared computation
- Incremental updates
- Detecting what to recompute


DON'T USE A SPREADSHEET! DO IT RIGHT.
BUT A SPREADSHEET WOULD BE SO EASY.
IN THE LONG RUN YOU'LL REGRET IT!
TAKE THE TIME TO WRITE REAL CODE.
JUST PASTE THE DATA! TINKER UNTIL IT WORKS!
BUILD A LABYRINTH OF REGEXREPLACE() AND ARRAYFORMULA()!
FEEL THE POWER!
FIGHT THE TEMPTATION!
EVER TRIED QUERY() IN GOOGLE SHEETS? IT LETS YOU TREAT A BLOCK OF CELLS LIKE A DATABASE AND RUN SQL QUERIES ON THEM.
DON'T LISTEN TO—
...WAIT. REALLY?
YES. AND LET ME TELL YOU ABOUT IMPORTHTML()...
OOOH...

Benchmarking Spreadsheet Systems

Practitioner / Ting Yu

Basic operations to improve - Yes or No

<table>
<tr><th>Operation</th><th>How</th><th>Y/N</th><th>Cost</th></tr>
<tr><td>Load</td><td>Load data on demand according to viewport.</td><td>YES</td><td>Dev Hour.</td></tr>
<tr><td>Find and Replace</td><td rowspan="2">Determine if formulae need recomputation.</td><td rowspan="2">YES</td><td rowspan="2">Dev Hour.</td></tr>
<tr><td>Sort</td></tr>
<tr><td>Conditional Formatting</td><td rowspan="4">Should not update formulae. Store formulae results.

Whether to store formulae results depends on formulae quantity relative to the sheet size as well as the amount of data manipulation. Either the system automatically makes a decision or hand the decision to users.</td><td rowspan="4">MAYBE</td><td rowspan="4">Storage cost either to users or to cloud providers.</td></tr>
<tr><td>Filter</td></tr>
<tr><td>Aggregate</td></tr>
<tr><td>Pivot Table</td></tr>
</table>

Optimizations to implement - Yes or No

Optimization	How	Y/N	Cost
Indexing	Inverted indexing for Find-and-Replace. Column indexing on all columns for Countif and Vlookup. But costly for storage. Extra costly given frequent data manipulations.	MAYBE	Indexing construction time. Storage cost. Only worthwhile
In Memory Layout	Columnar layout. Cache locality.	YES	Dev hour.
Shared Computation	Reuse formula subexpression in bulk computation. Might be caused by formula result not stored. But in one single batch operation, formula results should be cached..	YES	Dev hour.
Redundant Computation	Difficult to search for exactly formula. Not a real-world scenario.	NO	
Incremental Update	Same problem with formulae results storage.	NO	

Takeaway

Design perspective:

For basic operations, most problems involve the central design choice on formula: no storage on materialized formula results. This is a **trade-off between cost of storage and performance**.

To make a decision for the trade-off, understanding user context and scenario is critical. This can be automatically inferred or decided by the user.

Product perspective:

With mature product like spreadsheets, performance improvement is not critical to drive revenue, especially given most of the spreadsheets users deal with relatively small data and spreadsheets are sold as a office suite at an enterprise level.

There is a potential for spreadsheets to take market share of DBMS-based analysis scenarios if they can handle more data.

Thoughts on database-like spreadsheets

DBaaS, such as, Airtable... Such products work like a spreadsheets but are probably implemented similar to a database. They are web-based, full of APIs for best SaaS integration. But their goal is to help with workflow management.

Benchmarking Spreadsheet Systems

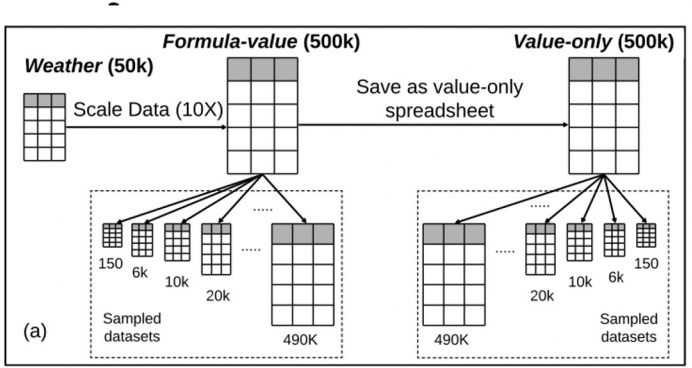
An Archeologist Perspective by: Cuong (Johnny) Nguyen

Paper Summary

- The paper wanted to benchmark spreadsheet systems for a variety of operations and workloads

- Came up with two benchmarks: Basic Complexity Testing (BCT) and Optimization Opportunities Testing (OOT)

- Found that all spreadsheets systems are only interactive for small datasets, one reason being they do not apply database-style optimizations



BCT	F	V
Data Load	√	√
Update	√	√
Select	√	√
Report	√	√
Aggregate	√	√
Lookup	X	√

(b)

OOT	F	V
Indexing	X	√
Materialization	X	√
Shared Computation	X	√

(c)

The Inspiration:

A Comparison of Approaches to Large-Scale Data Analysis by Pavlo et al. (2009)

- Cited by the paper as a good benchmarking paper in applying databases to large-scale data analytics, necessary to measure process + compare and contrast

- Found that while Hadoop does better than parallel DBMS in loading data, it is significantly slowing than parallel DBMS such as search, select, aggregate, join.

The Inspired

Efficient Specialized Spreadsheet Parsing for Data Science by Henze et al. (2022)

- Built off the paper's finding that spreadsheets system is extremely inefficient in data loading tasks, and that database-style optimizations should be applied to boost performance

- Introduces spreadsheet-specific optimizations to significantly reduce the runtime for loading massive spreadsheets by 2-3 times

Benchmarking Spreadsheet System

Academic Researcher Role : Vishnu K Krishnan

Georgia Tech

Paper Summary

- This paper aims to design a benchmarking system specifically for spreadsheets.
- The creation of 2 main benchmarks ensured wide coverage of operations - basic complexity testing (BCT) and Optimization opportunities testing (OOT).
- BCT - is used to test the limits of the spreadsheet based on normal operating conditions through the use of representative operators.
- OOT – is used to tests indexing-based optimization using querying operations like aggregate, report and lookup.
- First of its kind - present a benchmarking study for spreadsheets and compared performance of 3 popular spreadsheet systems.

Open Problems

- Lack of weightage for ease of use while measuring spreadsheets.
- Since there is not much previous work in benchmarking spreadsheets, it is tough to judge whether or not the metrics and values chosen are optimal.

Problems - Justification

- Spreadsheets are used commonly because of their relatively low skill training requirements. Measuring them based on this is only appropriate.

- A set of quantitative and qualitative guidelines set up can greatly help people determine the true characteristics of a given spreadsheet system.

Georgia Tech

Project Idea: Measure the Human in the spreadsheet

- Providing a model data-set and designing a set of questions or problems for the participants to solve using spreadsheets.(*Similar to the user study conducted in the hillview paper.)*
  - This will get us the required quantitative data
  - Conduct analysis on the data acquired to derive insights on the participants performance.

- Designing a set of interview questions or guidelines for them, to collect qualitative data on spreadsheets.
  - Example – which spreadsheet took you the least amount of time and why?

Georgia Tech

What has been done so far

- HillView: conducted an effectiveness case study.
  - Who ? - Uses operators familiar with Hillview for quantitative analysis – data scientists(Does not validate ease of use for beginners)
  - What ? - Designed set of problems to be solved targeting effective information extraction from data.
  - Why ? - To test functionality and usability of the product developed (HillView).

Georgia Tech

How to proceed

- Follow principles used in HillView to create problems for a generalized effectiveness study targeted towards spreadsheets.
- Conduct the study on a range of participant – beginner and intermediate to expert spreadsheet users.
- Incorporate qualitative data collection into benchmarking spreadsheet systems.
- Design evaluation process for the data collected and present findings.

Georgia Tech

Thank you!

Georgia Tech

How are spreadsheets different from DBMS?

Classic DB assumption:

Data systems should manage data in relations that can only be accessed through queries, that are unordered, and have a well-defined schema, with queries that operate on relations as a whole and kept separate from the data.

Spreadsheet:

- Data is ordered, and position is central
- Data can be directly manipulated
- Queries (Formula) are embedded as materialized views along with data
- Data is ad-hoc and cell-structured, not relational

Direct manipulation [Shneiderman'83]

Direct manipulation interfaces have four properties:

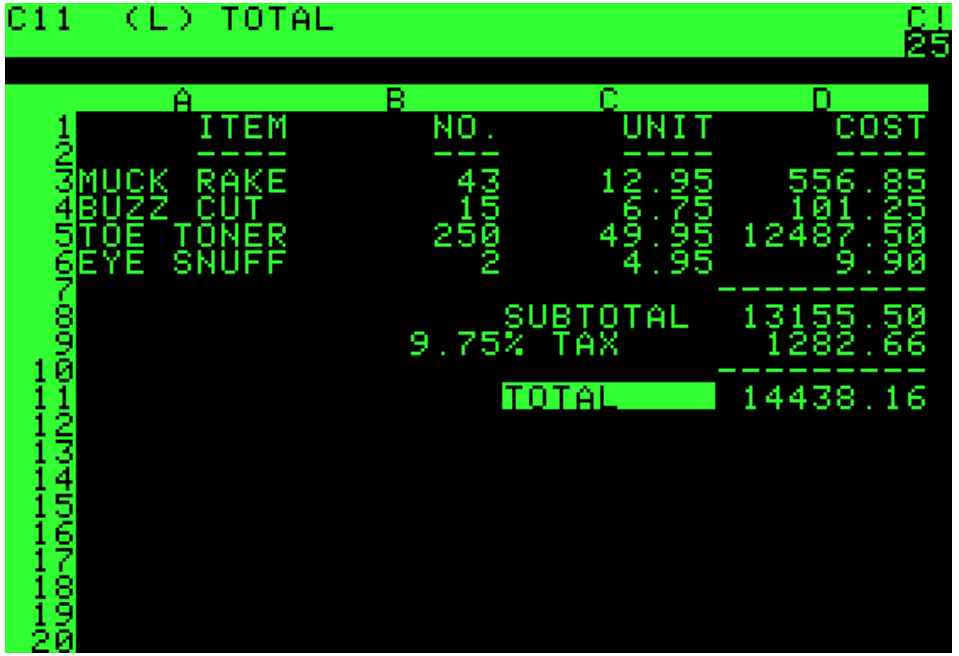
- Continuous representations of the objects and actions of interest
- Physical actions instead of complex syntax
- Continuous feedback and reversible, incremental actions
- Rapid Learning

Examples of direct manipulation in real life:

- driving a car via a steering wheel
- dragging a document to the trash
- inserting characters in a document by pointing to where they should go (with a mouse/cursor/insertion point) and then typing

Making an analogy

Build on user's existing experiences and intuitions to aid learning



Direct manipulation [Shneiderman'83]

Direct manipulation interfaces have four properties:

- Continuous representations of the objects and actions of interest
- Physical actions instead of complex syntax
- Continuous feedback and reversible, incremental actions
- Rapid Learning

Why are relational databases NOT direct manipulation interfaces?

What aspects of spreadsheets make them direct manipulation interfaces?

Benefits of Direct Manipulation

- While interacting with DM interfaces, users feel as if they are interacting with the domain rather than with the interface, so they focus on the task rather than on the technology. There is a feeling of direct involvement with a world of task objects rather than communication with an intermediary.
- Users can see the effects of their actions, and can change them if needed
- Users gain confidence and mastery because they are initiators of actions, they feel in control, and system responses are predictable

Disadvantage of Direct Manipulation

Continuous representations of the objects of interest

Can only act on a small number of objects that can be seen

Physical actions instead of complex syntax

Risk of RSI (repetitive strain injury)

Continuous feedback and reversible, incremental actions

Only if you attempt an operation that the system lets you do

Rapid Learning

Good for novice but repetitive tasks are not well supported

Contribution/Strengths

- First benchmark study of spreadsheet systems (It even does not have a Related Works section)
- Creative ways of probing the spreadsheet system
- Lots of very simple improvements can probably be made to the current spreadsheet systems. This paper opened up the scope for a lot of ideas.

Limitation/Weaknesses

- OOT benchmarking biases towards database solutions – It seems that the paper is suggesting more database style solutions with their benchmarking
- Desktop spreadsheet systems (Excel and Calc) and web-based spreadsheet systems (Google Sheets, Excel Online) are two quite different categories.
- Choice of spreadsheet systems: outdated versions, 2 closed sourced systems that need approximation

Limitation/Weaknesses

- .csv files might have different performances compared to .xls, .xlsx and .ods files (the software specific files)
- do not consider how collaborative editing affects Google Sheet
- The study does not talk about any software usage data. Maybe most users of Excel do not go above 10k rows, and if they don't what is the pain point for them at, say, 5k rows?
- Only one dataset (which was scaled up and down) was used for the entire study.

Limitation/Weaknesses

- Unclear how easy/difficult it is for future researchers to test black-box spreadsheet systems using the developed benchmark system
- Measure the performance of DBMS-backed spreadsheets (like Hillview).
- For the data load operation, the authors say 500ms is the interactivity expectation, although that seems like a pretty hard constraint for a one-time operation like data load.

Next class

Finding Related Tables in Data Lakes for Interactive Data Science

Author: Qiandong, Shen En

Reviewer: Vishnu

Archaeologist: Yanhao

Practioner: Haotian